Digital Millwork

from Carr, N. G. (2009). The big switch: Rewiring the world, from Edison to Google. W.W. Norton & Co.

CHAPTER 3

Digital Millwork

AS THE TWENTIETH century dawned, companies weren't just retooling their industrial machines to run on the electric current pumped out by utilities. They were also starting to install a very different type of electrical machine, a machine that processed information rather than material and that was operated by office clerks rather than factory hands. The machine, called a punch-card tabulator, had been invented in the early 1880s by an engineer named Herman Hollerith for the purpose of automating the US census. It worked on a simple principle. By punching holes in certain locations on a paper card, you could store information. A single census card, for instance, might hold all the data collected on one family. A hole in one place on the card would indicate that the family had three children, while a hole in another place would indicate that the family lived in an apartment. You could then place the card onto an electrically charged plate in Hollerith's machine and lower a grid of thin metal needles onto it. Wherever a hole was punched, a needle would pass through, completing a circuit and allowing the data on the card to be recorded on a meter. It was a binary system—either a hole was punched in a given location or it wasn't—which anticipated the binary operation of today's digital computers. More than that, though, the way the punch-card tabulator came to be sold and used would set the pattern for the entire modern history of business computing.

The Census Bureau put Hollerith's machine to work in the 1890 census, with great success. The tally proceeded much more quickly than it had in the 1880 round, even though the country's population had grown by about a quarter in the interim. The cost of the census was cut by $5 million—a savings almost ten times greater than the Bureau had expected. Having proved its value in speeding up calculations, the punch-card tabulator attracted the attention of the owners of large businesses like railroads, insurance agencies, banks, and mass-market manufacturers and retailers. As these companies had expanded their operations in the wake of the Industrial Revolution, they'd found it necessary to collect, store, and analyze ever larger amounts of data—on their customers, their finances, their employees, their inventories, and so on. Electrification allowed the companies to grow larger still, further expanding the information they had to process. This intellectual work became as important, and often as arduous, as the physical labor of manufacturing products and delivering services. Hollerith's tabulator allowed big businesses to process information far more quickly, with fewer people and greater accuracy, than was possible before.

Seeing the commercial potential of his invention, Hollerith established the Tabulating Machine Company to sell tabulators to businesses. The company grew rapidly, introducing a series of related products such as alphabetic tabulators, card sorters, card duplicators, and printers and selling them to an ever broader clientele. In 1911, Hollerith's firm merged with the Computer–Tabulating–Recording Company, an even larger supplier of business machines. A talented young manager named Thomas J. Watson was brought in to run the business. Thirteen

years later, the ambitious Watson changed the company's name to the more impressive sounding International Business Machines Corporation. Other companies, like Burroughs and Remington Rand in the United States and Bull in Europe, also rushed into the burgeoning punch-card market, competing with Watson's IBM.

The information technology industry had been born.

The spread of punch-card technology accelerated as standards were established for the design of the cards and the operation of the equipment and as technical advances and competition pushed down prices. Within a couple of decades, most large companies had set up punch-card rooms for the various machines they used to sort, tabulate, and store financial and other business information. They invested a great deal of capital in the machinery, and they hired clerks and technical specialists to operate and maintain it. They also developed close ties with the suppliers of the systems. "Processing information gathered into a deck of cards was entrenched into business practices by the mid-1930s," computer historian Paul Ceruzzi writes, "and reinforced by the deep penetration of the punched-card equipment salesmen into the accounting offices of their customers."

Even as companies were dismantling their power-generation departments, in other words, they were establishing new departments dedicated to the fledgling technology of automated data processing. During the second half of the century, these departments would grow dramatically as electronic digital computers replaced punch-card machines. Most big companies would find themselves building ever more elaborate installations of computing hardware and software, spending tens or even hundreds of millions of dollars a year on their in-house computing operations, and relying ever more heavily on IT vendors and consultants to keep their systems running. As the manipulation of symbols—words, numbers, images—supplanted the manipulation of materials as the focus of business in the developed world, the private power-generating plant of the nineteenth century found its echo in the private data-processing plant of the twentieth. And just as before, companies came to assume that there was no alternative—that running a complex computing operation was an intrinsic part of doing business.

ALTHOUGH IT NOW seems inevitable that the computer would become the mainstay of modern business, there was originally much skepticism about the machine's usefulness. When the first true commercial computer, the UNIVAC, was being built in the 1940s, few people believed it had much of a future in the corporate world. At the time, it was hard to imagine that many companies would have a need for the kind of intensive mathematical calculations that an electronic computer could churn out. The old punch-card tabulators seemed more than sufficient for handling transactions and keeping accounts. Howard Aiken, a distinguished Harvard mathematician and a member of the US government's National Research Council, dismissed as "foolishness" the idea that there would be a big market for computers. He believed that the country would need no more than a half dozen of them, mainly for military and scientific research. Even Thomas Watson is reputed to have said, in 1943, "I think there is a world market for about five computers."

But the designers of the UNIVAC, two University of Pennsylvania professors named J. Presper Eckert and John Mauchly, saw things differently. They realized that because an

electronic computer would be able to store its operating instructions in its own memory, it could be programmed to perform many functions. It wouldn't just be a glorified calculator, limited to preset mathematical routines. It would become a general purpose technology, a machine-of-all-trades, that companies could apply not just to everyday accounting chores but to innumerable managerial and analytical tasks. In a 1948 memo, Mauchly listed nearly two dozen companies, government agencies, and universities that he believed would be able to put the UNIVAC to good use. As it turned out, the market proved a good deal larger than even he expected.

Leading the way in adopting the powerful new machines was, once again, the Census Bureau. On March 31, 1951, it purchased the first UNIVAC, installing it a year later in its headquarters in Washington DC. By the end of 1954, Eckert and Mauchly's computers were running in the offices of ten private corporations, including General Electric, US Steel, Du Pont, Metropolitan Life, Westinghouse, and Consolidated Edison, the descendant of Thomas Edison's Edison Electric Illuminating Company. UNIVACs performed all the jobs that punch-card systems did—billing, payroll management, cost accounting—but they were also used for more complicated tasks like sales forecasting, factory scheduling, and inventory management. In short order, skepticism about the role of computers in business gave way to rampant enthusiasm. "The Utopia of automatic production is inherently plausible," the *Harvard Business Review* proclaimed in the summer of 1954.

The enthusiasm spread to the manufacturers of business machines, who saw in computing an expansive and lucrative new market. Soon after the UNIVAC appeared, IBM introduced its own line of mainframe computers, the 701 series, and by 1960 Honeywell, General Electric, RCA, NCR, Burroughs, and AT&T's Western Electric division were all in competition to sell computer gear. An entirely new industry—software programming—also began to take shape. About forty small software companies, with names like Computer Sciences Corporation, Computer Usage Company, and Computer Applications Inc., were founded during the late 1950s to write programs for mainframes.

It wasn't long before businesses were competing not only on the quality of their products but on the capabilities of their hardware and software. Once one company introduced a new system to automate an activity, other companies, fearful of being put at a disadvantage, followed suit. The first battles in what would become a worldwide information-technology arms race took place in the airline business. In 1959, Cyrus Rowlett Smith, the president of American Airlines, launched an ambitious project to build a system that would automate the making of flight reservations and the issuing of tickets, two labor-intensive processes that lay at the heart of the business. Built by 200 technicians over the course of more than five years, the system, called Sabre, incorporated two of IBM's most powerful mainframe computers as well as sixteen data-storage devices and more than 1,000 terminals for ticket agents. In addition to assembling the hardware, the project entailed the writing of a million lines of software code. When the system went into full operation at the end of 1965, it was able to process 40,000 reservations and 20,000 ticket sales a day—an astonishing feat at the time.

Sabre provided as great an advantage to American Airlines as Burden's waterwheel had provided to his ironworks. American was able to operate with fewer employees and higher productivity than other airlines, which continued to process reservations by hand. It also enjoyed big benefits in customer service, as it was able to respond to travelers' requests and inquiries far

more quickly than its rivals could. It gained an intelligence edge as well, as it could monitor demand for different routes and adjust ticket prices with great precision. Building and running computer systems had become as important to the success of American Airlines as flying planes and pampering passengers. In the years that followed, all the other major airlines, including Pan American, Delta, and United, built similar systems. They saw that they had no choice if they wanted to remain competitive. Not surprisingly, they found eager partners in computer vendors like IBM, Sperry Rand, and Burroughs, which earned big profits by replicating similar systems in one company after another.

Bank of America initiated a similar cycle of copycat investments in the banking industry when, in 1960, it unveiled its groundbreaking Electronic Recording Machine Accounting computer in a televised extravaganza hosted by Ronald Reagan. Within two years, the bank had thirty-two ERMA computers up and running, processing nearly five million checking and savings accounts that formerly had to be updated manually. The computers' ability to handle transactions with unprecedented speed and accuracy forced all major financial institutions to follow in Bank of America's footsteps. The same phenomenon was soon playing out in every other industry, as companies matched one another's investments in the latest computer gear.

But the mainframe era proved to be only the very start of the business world's great computer shopping spree. At the close of the 1960s, the average American company devoted less than 10 percent of its capital equipment budget to information technology. Thirty years later, that percentage had increased more than fourfold, to 45 percent, according to Commerce Department statistics. By 2000, in other words, the average US company was investing almost as much cash into computer systems as into all other types of equipment combined. Spending on software alone increased more than a hundred-fold during this period, from $1 billion in 1970 to $138 billion in 2000. The rest of the developed world saw a similar burst in investment, as global IT expenditures leapt from less than $100 billion a year in the early 1970s to more than $1 trillion a year by the early 2000s.

What happened during those thirty years? Businesses changed, and computers changed. As the economy became weighted more toward services and less toward manufacturing, investment shifted away from industrial machinery and into information technology. At the same time, computers themselves became smaller, cheaper, easier to program, and more powerful, dramatically expanding the range of tasks they could be applied to. Most important of all, computers became personal—they turned into commonplace tools that nearly all office employees could use.

During the mainframe era, by contrast, computers were institutional machines. Because it was so expensive to buy or lease a mainframe—the rent on a typical IBM computer was about $30,000 a month in the mid-1960s—a company had to keep the machine in constant use if it was to justify the expense. That meant that individual employees almost never had direct access to a computer. Like the punch-card tabulators that preceded them, mainframes and related gear were secluded in special rooms and operated by dedicated staffs of white-suited specialists—a "priesthood of technicians," in Ceruzzi's words. To use one of the machines, an employee had to store the program he wanted to run, along with all the necessary data, on a loop of tape or a deck of cards, and then put his job, or "batch," into a queue along with the jobs of his colleagues. The operators of the mainframe would run one batch after another, bundling the results into printouts that employees would then pick up and review. If an employee found an error, he would have to

resubmit his batch and go through the entire cycle again.

Mainframe batch-processing had one big advantage: it ensured that a computer was used efficiently. No machine sat idle, at least not for long. The typical corporate mainframe operated at more than 90 percent of its full capacity. But batch-processing had an even larger disadvantage: it rendered computing impersonal. The organizational and technological barriers that stood between the employee and the machine stifled experimentation and constrained the application of computing power, while the delay in receiving results prevented computers from being used to support the many small, everyday decisions necessary to keep a company running.

This shortcoming didn't last long. As technological innovation sped forward through the 1960s and 1970s, computers shrank in size and price. Tiny transistors replaced bulky vacuum tubes, and cheap standardized components replaced expensive custom parts, giving rise to relatively inexpensive minicomputers that could fit beside a desk. Minicomputers didn't supplant mainframes; they supplemented the bigger, more powerful machines. But they greatly expanded businesses' use of computers. Because minicomputers could be connected to desktop terminals, they allowed ordinary employees to tap directly into the power of a computer to do a wide range of jobs, from analyzing business investments to designing new products to scheduling assembly lines to writing letters and reports. The languages for writing software also became much simpler during this time. Programmers could write code using basic English words and syntax rather than long lines of numbers. That greatly expanded the computer programming industry, bringing a big jump in the number of programmers and in the kinds of applications they developed. By the early 1970s, a company could buy a minicomputer for less than $10,000 and quickly program it to carry out a specialized task.

The minicomputer business flourished, propelling companies like Digital Equipment, Wang, and Apollo into the front ranks of the computer industry. But its heyday proved brief. The mini turned out to be a transitional machine. Breakthroughs in the design of integrated circuits, particularly the invention of the microprocessor by Intel engineers in 1971, led to the introduction and rapid proliferation of an entirely new kind of machine—the micro, or personal, computer—that was even smaller, even cheaper, and even easier to use than the mini. The arrival of the microcomputer would quickly upend the industry, ushering in a new era in business computing.

AS WITH MAINFRAMES, experts didn't at first see much potential for personal computers in business. This time, though, the doubts were of a very different sort. Where mainframes were viewed as too powerful for commercial applications, PCs were seen as too weak. They were dismissed as frivolous gadgets, toys for geeky hobbyists. The dominant computer companies of the day, from IBM to Digital, paid the quirky new machines little mind. It took a college dropout named Bill Gates—himself a geeky hobbyist—to see the potential of personal computers in business. In 1975, Gates and his high-school buddy Paul Allen founded a little company named Micro-Soft to write software for the newly invented PC. Gates soon realized that the machine would not only find a place inside business but that, because of its versatility and low cost, it would supplant the mainframe as the center of corporate computing. Any company able to gain control over the PC's operating system, and the virtual desktop it created, would become the most powerful in the computing business. Gates's insight would

make Microsoft—as he renamed the company—the dominant player in the IT industry and bring Gates himself unimaginable wealth.

The PC democratized computing. It liberated the computer from corporate data centers and IT departments and turned it into a universal business tool. In the process, it also changed the way companies organized their computing assets and operations. The personal computers sitting on office workers' desks soon came to be wired together into networks to allow employees to exchange files and share printers. The old centralized mainframe rooms didn't disappear. They were transformed into a new kind of data center. The engine rooms of modern businesses, these centers contained the storage systems that held companies' most important data as well as the powerful server computers that ran the applications used to manage their finances and operations. Individual employees could use their PCs to run their own personal programs, such as Microsoft Word and Excel, but they could also use them to tap into the programs and files on the central servers. Because the PCs acted as "clients" of the shared servers, this setup came to be known as "client–server computing." It became the defining model of corporate computing in the PC age, the model that has remained dominant to the present day.

Client–server computing has turned out to be the mirror image of mainframe computing. It has made computing personal, but it has also made it woefully inefficient. Corporate computer systems and networks, the digital millwork of the modern company, have become steadily more complex as their applications have multiplied. One of the main reasons for the complexity is the historical lack of standards in computing hardware and software. Vendors have tended to promote their own proprietary products, which by design don't mesh well with competitors' gear. As a result, corporate software programs have generally been written to run on a particular operating system, a particular microchip, a particular database, and a particular hardware setup. Unlike the multipurpose mainframes, most server computers have had to be used as single-purpose machines, dedicated to running just one software application or one database. Whenever a company buys or writes a new application, it has to purchase and install another set of dedicated computers. Each of these computers, moreover, has to be configured to handle the peak theoretical demand for the application it runs—even if the peak load is rarely or ever reached.

The proliferation of single-purpose systems has resulted in extraordinarily low levels of capacity utilization. One recent study of six corporate data centers revealed that most of their 1,000 servers were using less than a quarter of their available processing power. Other studies indicate that data storage systems are almost equally underused, with capacity utilization averaging between 25 and 50 percent. Before the PC age, data-processing professionals viewed the conservation of computing resources as not just an economic imperative but an ethical one. "To waste a CPU cycle or a byte of memory was an embarrassing lapse," recalls the science writer Brian Hayes. "To clobber a small problem with a big computer was considered tasteless, and unsporting, like trout fishing with dynamite." The client–server model killed the conservation ethic. Profligacy replaced frugality as the defining characteristic of business computing.

The complexity and inefficiency of the client–server model have fed on themselves over the last quarter century. As companies continue to add more applications, they have to expand their data centers, install new machines, reprogram old ones, and hire ever larger numbers of technicians to keep everything running. When you also take into account that businesses have to

buy backup equipment in case a server or storage system fails, you realize that, as studies indicate, most of the many trillions of dollars that companies have invested into information technology have gone to waste.

And there are other costs as well. As data centers have expanded and become more densely packed with computers, electricity consumption has skyrocketed. According to a December 2005 study by the Department of Energy's Lawrence Berkeley National Laboratory, a modern corporate data center "can use up to 100 times as much energy per square foot as a typical office building." The researchers found that a company can spend upwards of $1 million per month on the electricity required to run a single big data center. And the electric bill continues to mount rapidly as servers proliferate and computer chips become more powerful and power-hungry. Luiz André Barroso, a computer engineer with Google, concludes that, barring substantial improvements in the efficiency of computers, "over the next few years, power costs could easily overtake hardware costs, possibly by a large margin."

The waste inherent in client–server computing is onerous for individual companies. But the picture gets worse—much worse—when you look at entire industries. Most of the software and almost all of the hardware that companies use today are essentially the same as the hardware and software their competitors use. Computers, storage systems, networking gear, and most widely used applications have all become commodities from the standpoint of the businesses that buy them. They don't distinguish one company from the next. The same goes for the employees who staff IT departments. Most perform routine maintenance chores—exactly the same tasks that their counterparts in other companies carry out. The replication of tens of thousands of independent data centers, all using similar hardware, running similar software, and employing similar kinds of workers, has imposed severe penalties on the economy. It has led to the overbuilding of IT assets in almost every sector of industry, dampening the productivity gains that can spring from computer automation.

The leading IT vendors have ridden the investment wave to become some of the world's fastest-growing and most profitable businesses. Bill Gates's company is a perfect case in point. Almost every company of any size today buys copies of Microsoft Windows and Microsoft Office for all its white-collar workers, installing the software individually on every PC and upgrading the programs routinely. Most also run at least some of their servers on a version of the Windows operating system and install other expensive Microsoft programs in their data centers, such as the Exchange software used to manage email systems. In the three decades since its founding, Microsoft grew to have annual sales of nearly $50 billion, annual profits of more than $12 billion, and more than $30 billion of cash in the bank. And Microsoft has plenty of company, from other software makers like Oracle and SAP to server suppliers like IBM and Hewlett–Packard to PC vendors like Dell to the hundreds of consulting firms that feed off the complexity of modern business computing. They've all happily played the role of weapons suppliers in the IT arms race.

WHY HAS COMPUTING progressed in such a seemingly dysfunctional way? Why has the personalization of computers been accompanied by such complexity and waste? The reason is fairly simple. It comes down to two laws. The first and most famous was formulated in 1965 by the brilliant Intel engineer Gordon Moore. Moore's Law says that the power of

microprocessors doubles every year or two. The second was proposed in the 1990s by Moore's equally distinguished colleague Andy Grove. Grove's Law says that telecommunications bandwidth doubles only every century. Grove intended his "law" more as a criticism of what he considered a moribund telephone industry than as a statement of technological fact, but it nevertheless expresses a basic truth: throughout the history of computing, processing power has expanded far more rapidly than the capacity of communication networks. This discrepancy has meant that a company can only reap the benefits of advanced computers if it installs them in its own offices and hooks them into its own local network. As with electricity in the time of direct-current systems, there's been no practical way to transport computing power efficiently over great distances.

As Grove's observation reveals, the scarcity of communications bandwidth has long been recognized as a barrier to effective and efficient computing. It has always been understood that, in theory, computing power, like electric power, could be provided over a grid from large-scale utilities—and that such centralized dynamos would be able to operate much more efficiently and flexibly than scattered, private data centers. Back in 1961, when computer scientists were just beginning to figure out how to get computers to talk to each other, one expert in the fledgling art of networking, John McCarthy, predicted that "computing may someday be organized as a public utility, just as the telephone system is organized as a public utility." Every advance in networking brought a new wave of entrepreneurs hoping to turn utility computing into a big business. In the mainframe era, time-sharing companies set up central computers and rented them out to other businesses, allowing direct connections over phone lines. In the 1970s, companies like Automated Data Processing began to provide some routine computing jobs—payroll processing, notably—as for-fee services. And in the 1990s, a slew of "application service providers" emerged, with considerable venture-capital backing, in hopes of providing businesses with software programs over the Internet.

But all these attempts at utility computing were either doomed or hamstrung by the lack of sufficient bandwidth. Even during the late 1990s, as the telecommunications industry rushed to modernize its networks, broadband capacity was still neither inexpensive nor plentiful enough for utilities to deliver computing services with the speed and reliability that businesses enjoyed with their local machines. And so companies continued to entangle themselves in their digital millwork, accepting complexity, inefficiency, and waste as the costs of automating their operations.

But now, at last, that's changing. The network barrier has, in just the last few years, begun to collapse. Thanks to all the fiber-optic cable laid by communications companies during the dotcom boom—enough, according to one estimate, to circle the globe more than 11,000 times—Internet bandwidth has become abundant and abundantly cheap. Grove's Law has been repealed. And that, when it comes to computing at least, changes everything. Now that data can stream through the Internet at the speed of light, the full power of computers can finally be delivered to users from afar. It doesn't matter much whether the server computer running your program is in the data center down the hall or in somebody else's data center on the other side of the country. All the machines are now connected and shared—they're one machine. As Google's chief executive, Eric Schmidt, predicted way back in 1993, when he was the chief technology officer with Sun Microsystems, "When the network becomes as fast as the processor, the computer hollows out and spreads across the network."

What the fiber-optic Internet does for computing is exactly what the alternating-current network did for electricity: it makes the location of the equipment unimportant to the user. But it does more than that. Because the Internet has been designed to accommodate any type of computer and any form of digital information, it also plays the role of Insull's rotary converter: it allows disparate and formerly incompatible machines to operate together as a single system. It creates harmony out of cacophony. By providing a universal medium for data transmission and translation, the Net is spurring the creation of centralized computing plants that can serve thousands or millions of customers simultaneously. What companies used to have no choice but to supply themselves, they can now purchase as a service for a simple fee. And that means they can finally free themselves from their digital millwork.

It will take many years for the utility computing system to mature. Like Edison and Insull before them, the pioneers of the new industry will face difficult business and technical challenges. They'll need to figure out the best ways to meter and set prices for different kinds of services. They'll need to become more adept at balancing loads and managing diversity factors as demand grows. They'll need to work with governments to establish effective regulatory regimes. They'll need to achieve new levels of security, reliability, and efficiency. Most daunting of all, they'll need to convince big companies to give up control over their private systems and begin to dismantle the data centers into which they've plowed so much money. But these challenges will be met just as they were met before. The economics of computing have changed, and it's the new economics that are now guiding progress. The PC age is giving way to a new era: the utility age.